

# 证 明

本证明之附件是向本局提交的下列专利申请副本

申 请 日： 2003. 11. 13

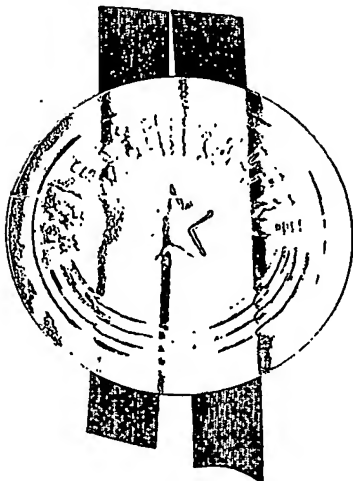
申 请 号： 2003101136049

申 请 类 别： 发明

发明创造名称： 一种基于辨群共轭问题的数字签名方法

申 请 人： 中兴通讯股份有限公司

发明人或设计人： 丁勇、陈剑勇、李亚晖



中华人民共和国  
国家知识产权局局长

王 崇 川

2004 年 12 月 8 日

BEST AVAILABLE COPY

# 权利要求书

1. 一种基于辨群共轭问题的数字签名方法，其特征在于，所述方法包括以下处理过程：

5 签名方为 S，签名验证方为 V，需要签名的消息为 m，系统参数：辨群  $B_n(l)$ ， $B_n(l)$  的左子群  $LB_m(l)$ ， $B_n(l)$  的右子群  $RB_{n-1-m}(l)$ ，辨群公钥对间的距离 d，从比特串  $\{0, 1\}^*$  到辨群  $B_n(l)$  的单向散列函数 h，用于 CDP 判定算法 BCDA 的素数 p 和判定的点的个数 r；

签名方对需要签名的消息进行如下操作：

- 1) 签名方 S 随机生成一个辨元  $x \in LB_m(l)$ ；
- 10 2) 签名方 S 使用  $RSSBG(x, d)$  生成  $(x', a) \in B_n(l) \times B_n(l)$ ，使得  $x' = a^{-1}xa$ ；将共轭对  $(x', x)$  作为 S 的公钥，共轭元 a 作为 S 的私钥；
- 3) 签名方 S 对需要签名的消息 m 首先使用散列函数 h 得到  $y = h(m) \in B_n(l)$ ，然后随机生成一个辨元  $b \in RB_{n-1-m}(l)$ ，然后使用自己的私钥 a 和产生的随机辨元 b 对消息 m 签名得到  $Sign(m) = a^{-1}byb^{-1}a$ ；
- 15 4) 签名者将消息 m、公钥  $(x', x)$ 、m 的签名  $Sign(m)$  发送给签名验证方 V；

签名验证方 V 收到签名者发送的信息后进行如下操作：

- 5) 首先利用系统参数散列函数 h 对消息 m 作用计算得到  $y = h(m)$ ；
- 6) 判定  $sign(m) \sim y$  是否成立，若不成立，则  $sign(m)$  不是一个合法签名；若成立，则转步骤 7)；
- 20 7) 判定  $sign(m)x' \sim xy$  是否成立，若不成立，则  $sign(m)$  不是一个合法签名；若成立，则  $sign(m)$  为消息 m 的合法签名。

2. 根据权利要求 1 所述的基于辨群共轭问题的数字签名方法，其特征在于，所述步骤 2) 生成公钥和私钥具体包括以下处理过程：

- 25 (2-1) 选定系统参数  $n, m, l, d$ ；
- (2-2) 随机生成一个辨元 x 属于集合  $LB_m(l)$ ；
- (2-3) 随机选择一个辨元 b 属于集合  $B_n(5l)$ ；
- (2-4) 计算  $y = b^{-1}xb$ ；
- (2-5) 随机产生一个比特，如为 0，则计算  $y = \text{cycling}(y) = a^{-1}xa$ ，否则计算  $y = \text{decycling}(y) = a^{-1}xa$ ；
- 30

(2-6) 判断  $y$  是否属于集合  $SSS(x)$  以及  $l(y) \leq d$  是否都成立, 若都成立则输出  $(x, y)$  为公钥,  $a$  为私钥; 若有一个不成立, 则转入 (2-5) 进一步生成。

3. 根据权利要求 1 所述的基于辫群共轭问题的数字签名方法, 其特征在于, 所述散列矩阵的处理过程是:

(3-1) 选择一个普通散列函数  $H$ , 其输出  $H(m)$  长度为  $l[\log(2, n!)]$  (可以截取), 然后将  $H(m)$  一次等分为  $l$  段  $R_1 || R_2 || \dots || R_l$ ;

(3-2) 一次将  $R_i$  对应为置换辫元  $A_i$ , 然后计算  $h(m) = A_1 * A_2 \dots A_l$  即为所求的  $h(m)$ 。

4. 根据权利要求 1 所述的基于辫群共轭问题的数字签名方法, 其特征在于, 所述的步骤 6)、7) 中利用算法 BCDA 判定共轭的处理过程是:

(4-1) 输入辫元  $a, b$ , 选择好系统参数  $p, r$  并计算其特征  $Pa(t)$  以及  $Pb(t)$ ;

(4-2) 随机选择  $r$  个不同的随机数  $t_i (i=1, 2 \dots r)$ , 并计算  $Pa(t_i) = Pb(t_i)$  都成立; 若有一个不成立, 则  $a \sim b$  不成立; 若都成立, 则转 (4-3);

(4-3) 计算  $\text{Maxinf}(a) = \text{Maxinf}(b)$  是否成立, 若不成立, 则  $a \sim b$  不成立; 若成立, 则转 (4-4);

(4-4) 计算  $\text{Minsup}(a) = \text{Minsup}(b)$  是否成立, 若不成立, 则  $a \sim b$  不成立; 若成立, 则转则  $a \sim b$  不成立。

5. 根据权利要求 1 所述的基于辫群共轭问题的数字签名方法, 其特征在于, 所述  $n$  取  $20 \sim 30$  间,  $l=3$ ,  $d=4$ ,  $p$  为  $2^{31} \sim 2^{32}$  间, 以及左辫群大小  $m$  比  $n$  小 4。

## 一种基于辫群共轭问题的数字签名方法

### 技术领域

5 本发明涉及一种基于辫群的 CSP（共轭搜索）问题和 CDP（共轭判定）问题间差异的数字签名方法 ECSS。本发明涉及到信息安全领域具体说就是签名者如何发布一个带有自己私钥签名文件以便验证者使用签名者的公钥来验证该文件是否为签名者发布的签名文件。

### 背景技术

10 现目前广泛使用的数字签名技术是 RSA 签名体制，它的安全性是建立在大数分解的困难性上的，然而随着计算机处理能力的不断提高和相关的研究逐渐深入，RSA 不得不不断的加大模数  $n$  位数以确保安全性，从 512 比特到 1024 比特到 2048 比特。由于密钥的位数过长，导致产生大素数和指数计算的计算量都很大，因此 RSA 的效率不是很高；而如果为了提高效率采用硬件实现，则位数过长导致设备更复杂、成本更高，  
15 而且由于硬件实现方式的不可改性，硬件的使用寿命缩短，导致成本的进一步提高。

自从 2000 年韩国学者 Ki Hyoung KO、Sang Jin Lee 在 CRYPTO 2000 提出了一种基于辫群共轭问题的困难性的密钥交换协议和公钥加密体制以来 (K. H. Ko, S. J. Lee, J. H. Cheon, J. W. Han, J. S. Kang and C. S. Park, *New Public-Key Cryptosystem Using Braid Groups*, *Proc. of Crypto 2000*, LNCS 1880, Springer-Verlag  
20 (2000) 166 ■ 183.)，辫群公钥密码体制得到了广泛的研究。然而它的数字签名体制一直没有一个很好的解决方案。直到 2003 年，韩国学者 Ki Hyoung KO、Doo Ho Cho 提出并实现了两种基于辫群共轭问题的签名体制 (Ki Hyoung Ko and Doo Ho Choi and Mi Sung Cho and Jang Won Lee *New Signature Scheme Using Conjugacy Problem* *Cryptology ePrint Archive: Complete Contents 2003/168*) SCSS 以及 CSS。  
25 我们简要介绍一下【3】中的两种签名体制，SCSS 以及 CSS。

#### 简单共轭签名体制 SCSS:

公共参数: 辫群  $B_n$  散列函数  $h$

密钥生成: 公钥: 一个 CSP 问题为困难问题的共轭对  $(x, \log_2^n x') \in B_n \times B_n$ ,

私钥:  $a \in B_n$ , 满足  $x' = a^{-1}xa$ ;

30 签名: 对于一个给定的比特串消息  $m$ ,  $m$  的签名  $\text{sign}(m) = a^{-1}ya$ , 其中  $y = h(m)$ 。

验证：一个签名  $\text{sign}(m)$  是合法的当且仅当： $\text{sign}(m) \sim y$ ,  $x' \text{sign}(m) \sim xy$   
然而由于攻击者可以获得很多对的  $(y_i, a^{-1}y_i a)$ , 从而可能造成私钥  $a$  的秘密信息泄漏, 即  $k$ -CSP 问题。为了克服以上问题, 他们提出了 CSS 签名体制。

共轭签名体制 CSS:

5 公共参数: 群  $B_n$  散列函数  $h$

密钥生成: 公钥: 一个 CSP 问题为困难问题的共轭对  $(xx') \in B_n \times B_n$ ,

私钥:  $a \in B_n$ , 满足  $x' = a^{-1}xa$ ;

签名: 对于一个给定的消息  $m$ , 随机选择一个随机化因子  $b \in B_n$ , 计算  $\alpha = b^{-1}xb$ ,  $y = h(m || \alpha)$ ,  $\beta = b^{-1}yb$ ,  $\gamma = b^{-1}ay a^{-1}b$ , 消息  $m$  的签名  $\text{sign}(m) = (\alpha, \beta, \gamma)$ 。

10 验证: 消息  $m$  的签名  $\text{sign}(m) = (\alpha, \beta, \gamma)$  为合法签名当且仅当满足:  $\alpha \sim x$ ,  $\beta \sim \gamma \sim y$ ,  $\alpha \beta \sim xy$ ,  $\alpha \gamma \sim x' y$

CSS 签名体制由于引入了随机化因子  $b$ , 因此很好的克服了  $k$ -CSP 问题。但是可以发现, 由于增加了更多的计算和数据, 因此整个效率明显下降。

15

#### 发明内容

本发明所要解决的技术问题是: 为了克服了现有技术中的容易受到针对大数分解攻击以及产生大素数消耗计算资源过大的缺点, 解决现有技术中存在的产生密钥和签名验证时间过长的问题, 提供一种基于群共轭问题的高效数字签名方法, 不但可以克服存在 SCSS 中的  $k$ -CSP 问题, 而且相对 CSS, 计算量和数据都将大大的减少, 提高了整个签名体制的效率。

本发明所述的基于群共轭问题的数字签名方法, 包含以下步骤:

25 假设签名方为  $S$ , 签名验证方为  $V$ , 需要签名的消息为  $m$ , 系统参数: 群  $B_n(l)$ ,  $B_n(l)$  的左子群  $LB_m(l)$ ,  $B_n(l)$  的右子群  $RB_{n-1-m}(l)$ , 群公钥对间的距离  $d$ , 从比特串  $\{0, 1\}^*$  到群  $B_n(l)$  的单向散列函数  $h$ , 用于 CDP 判定算法 BCDA 的素数  $p$  和判定的点的个数  $r$ 。

首先签名方对需要签名的消息进行如下操作:

1) 签名方  $S$  随机生成一个群元  $x \in LB_m(l)$ ;

2) 签名方  $S$  使用  $RSSBG(x, d)$  生成  $(x', x, a) \in B_n(l) \times LB_m(l) \times B_n(l)$ , 使得  $x' = a^{-1}xa$ ;

30 将共轭对  $(x', x)$  作为  $S$  的公钥, 共轭元  $a$  作为  $S$  的私钥;

$RSSBG(x, d)$  算法简单描述如下:

(2-1) 选定系统参数  $n, m, l, d$ 。

(2-2) 随机生成一个辨元  $x$  属于集合  $LB_m(l)$ .

(2-3) 随机选择一个辨元  $b$  属于集合  $B_n(5l)$ .

(2-4) 计算  $y = b^{-1}xb$

(2-5) 随机产生一个比特, 如为 0, 则计算  $y = \text{cycling}(y) = a^{-1}xa$ , 否则计算  $y =$

5  $\text{decycling}(y) = a^{-1}xa$

(2-6) 判断  $y$  是否属于集合  $SSS(x)$  以及  $l(y) \leq d$  是否都成立, 若都成立则输出

$(x, y)$  为公钥,  $a$  为私钥; 若有一个不成立, 则转入 (2-5) 进一步生成。

3) 签名方  $S$  对需要签名的消息  $m$  首先使用散列函数  $h$  得到  $y = h(m) \in B_n(l)$ , 然后随机生成一个辨元  $b \in RB_{n-1-m}(l)$ , 然后使用自己的私钥  $a$  和产生的随机辨元  $b$  对消息  $m$  签名得到  $\text{Sign}(m) = a^{-1}byb^{-1}a$ ;

10

有  $m$  散列得到  $h(m)$  的 成立流程简单描述如下:

(3-1) 选择一个普通散列函数  $H$ , 其输出  $H(m)$  长度为  $l[\log_2^n]$  (可以截取), 然后将  $H(m)$  一次等分为 1 段  $R1||R2|| \cdots ||Rl$ .

(3-2) 一次将  $R_i$  对应为置换辨元  $A_i$ , 然后计算  $h(m) = A1 * A2 \cdots A_l$  即为所求的  $h(m)$ .

15 4) 签名者将消息  $m$ 、公钥  $(x', x)$ 、 $m$  的签名  $\text{Sign}(m)$  发送给签名验证方  $V$ ;

签名验证方  $V$  收到签名者发送的信息后进行如下操作:

5) 首先利用系统参数散列函数  $h$  对消息  $m$  作用计算得到  $y = h(m)$ ;

6) 判定  $\text{sign}(m) \sim y$  是否成立, 若不成立, 则  $\text{sign}(m)$  不是一个合法签名; 若成立, 则转步骤 7);

20 7) 判定  $\text{sign}(m) x' \sim xy$  是否成立, 若不成立, 则  $\text{sign}(m)$  不是一个合法签名; 若成立, 则  $\text{sign}(m)$  为消息  $m$  的合法签名。

判定两个辨元  $a, b$  是否共轭的判定算法 BCDA 简单叙述如下:

(6-1) 输入辨元  $a, b$ , 选择好系统参数  $p, r$  并计算其特征  $Pa(t)$  以及  $Pb(t)$ 。

25 (6-2) 随机选择  $r$  个不同的随机数  $t_i (i=1, 2 \cdots r)$ , 并计算  $Pa(t_i) = Pb(t_i)$  都成立。若有一个不成立, 则  $a \sim b$  不成立; 若都成立, 则转 (6-3)。

(6-3) 计算  $\text{Maxinf}(a) = \text{Maxinf}(b)$  是否成立, 若不成立, 则  $a \sim b$  不成立; 若成立, 则转 (6-4)。

(6-4) 计算  $\text{Minsup}(a) = \text{Minsup}(b)$  是否成立, 若不成立, 则  $a \sim b$  不成立; 若成立, 则转则  $a \sim b$  不成立。

30 由于本发明的数字签名方法, 有如下优点:

1 由于加入了随机化因子  $b$ , 使得针对每个消息  $m$ , 共轭对  $(\text{sign}(m), h(m))$  的共轭元

为  $b^{-1}a$ ，因此每次的共轭元都不相同，从而掩盖了是要  $a$  的信息泄漏，避免了在现有技术中 SCSS（简单共轭签名体制）签名体制中单一使用私钥  $a$  作为共轭对  $(\text{sign}(m), h(m))$  的共轭元的  $k$ -CSP 问题。可参看表 1。

- 2 本方法相对于现有技术中的 CSS（共轭签名体制）数字签名方法在不降低安全性的前提下，大大节省了计算时间，提高了效率，可参看表 1。
- 5

| 签名体制  | 签名计算量                | 验证计算量                            | 签名数据量 | 安全性                                   |
|-------|----------------------|----------------------------------|-------|---------------------------------------|
| SCSS  | 共轭计算：1 次<br>散列计算：1 次 | 共轭判定：2 次<br>散列计算：1 次<br>辨群运算：2 次 | 1 个辨元 | 存在 $k$ -CSP 问题，安全性较低，基于 MCSP 问题       |
| CSS   | 共轭计算：4 次<br>散列计算：1 次 | 共轭判定：5 次<br>散列计算：1 次<br>辨群运算：4 次 | 3 个辨元 | 引入随机化密钥因子，解决了 $k$ -CSP 问题，基于 MTSP 问题  |
| 本发明方法 | 共轭计算：2 次<br>散列计算：1 次 | 共轭判定：2 次<br>散列计算：1 次<br>辨群运算：2 次 | 1 个辨元 | 引入随机化密钥因子，解决了 $k$ -CSP 问题，基于 MCSP 问题。 |

表 1 3 种签名方法比较

- 3 本发明相对于传统的 RSA 签名方法使用完全不同体系的数学基础，不需要产生大素数，大大节省了密钥的位数和签字的位数，节约了计算资源，提高了签名验证小效率。在现有技术中给出的 CSS 签名方法在 Pentium III 866MHz 处理器上得到的数据为表 2 所示（默认设置的参数  $l=3$ ,  $d=4$ ,  $2^{31} < p < 2^{32}$ ,  $r=3$ ）；
- 10

| n  | 公钥位数 | 签名位数 | 密钥生成时间   | 签名时间     | 验证时间     | 安全强度      |
|----|------|------|----------|----------|----------|-----------|
| 20 | 370  | 1653 | 17.82 ms | 18.68 ms | 30.87 ms | $2^{220}$ |

|    |     |      |          |          |          |           |
|----|-----|------|----------|----------|----------|-----------|
| 24 | 478 | 2138 | 21.70 ms | 22.79 ms | 41.75 ms | $2^{356}$ |
| 28 | 591 | 2648 | 24.42 ms | 25.77 ms | 59.59 ms | $2^{530}$ |

表2 CSS签名方法的试验数据表

而本方法相对于CSS签名方法签名时间和验证时间都将大大减少，因此相对RSA有效率更高的优点。

## 5 附图说明

图1 本发明基于辨群共轭问题的数字签名方法的步骤主流程图。

图2 基于辨群共轭问题的数字签名方法的密钥生成的子流程图。

图3 单向散列函数h的处理流程图。

图4 CDP问题判定算法BCDA处理流程图。

10 图5 签名方对消息m签名的流程图。

图6 验证方对消息签名的验证流程图。

## 具体实施方式

由于本发明涉及到一系列的数学原理，首先将本发明的数学背景阐述如下：

15 辨群  $B_n$  ( $n$  为群参数) 是由 Artin 生成元  $\sigma_1, \sigma_2, \dots, \sigma_{n-1}$  生成的有限表示的无限群，并且它的生成元  $\sigma_1, \sigma_2, \dots, \sigma_{n-1}$  满足以下关系：

$$\sigma_i \sigma_j = \sigma_j \sigma_i \quad (|i-j| > 1, \quad 1 < i, j < n) \quad (1)$$

$$\sigma_i \sigma_j \sigma_i = \sigma_j \sigma_i \sigma_j \quad (|i-j| > 1, \quad 1 < i, j < n) \quad (2)$$

由左边  $m$  个生成元  $\sigma_1, \sigma_2, \dots, \sigma_m$  生成的群叫  $B_n$  的左子群，记做  $LB_m$ ；而由右边的  $n-1-m$  个生成元  $\sigma_{m+1}, \sigma_{m+2}, \dots, \sigma_{n-1}$  生成的子群叫  $B_n$  的右子群，记做  $RB_{n-1-m}$ 。由生成元关系 (1) 显然可知：任取  $(x, y) \in LB_m \times RB_{n-1-m}$ ，有  $xy = yx$ 。对于一个辨元  $b$ ，若他只包含  $\sigma_1, \sigma_2, \dots, \sigma_{n-1}$  而不含  $\sigma_1, \sigma_2, \dots, \sigma_{n-1}$  的逆元，则称  $b$  为一个正元。若对于正元  $b, a$ ，有一个正元或单位元  $c$  使得  $b = ac$ ，则称  $a$  为  $b$  的子元。辨元  $\Delta = (\sigma_1 \sigma_2 \dots \sigma_{n-1}) (\sigma_1 \sigma_2 \dots \sigma_{n-2}) \dots (\sigma_1 \sigma_2) (\sigma_1)$  称为辨群  $B_n$  的本元。  $\Delta$  满足  $\Delta b = \tau(b) \Delta$ ， $\tau(\sigma_i) = \sigma_{n-i}$ 。其中  $\Delta$  的子元称作置换元，他们组成的集合和对称群  $\Sigma_n$  的  $n!$  个元素一一对应。因此  $\Delta$  的子元可用一个置换  $\pi : \{0, 1, \dots, n-1\} \rightarrow \{0, 1, \dots, n-1\}$  来表示。任何一个辨元  $b$  都存在唯一的一个标准表示形式： $b = \Delta^u \pi_1 \pi_2 \dots \pi_l$ ，其中  $\pi_i$  为一个置换元。  $b$  的几个长度定义如下： $\inf(b) = u$ ， $\sup(b) = u + l$ ， $l(b) = l$ 。

30 在一个辨群  $B_n$  中，如果对于两个辨元  $x, y \in B_n$ ，存在一个辨元  $a \in B_n$  使得  $y = a^{-1}xa$ ，则称辨元  $x, y$  共轭，记做  $x \sim y$ ，而辨元  $a$  称作共轭对  $(x, y)$  的共轭元，显然 “ $\sim$ ”



是一种等价关系。辨群的基本共轭问题包括共轭判定问题 CDP 问题和共轭元搜索问题 CSP 问题。所谓 CDP 问题就是指：任意给出辨元对  $(x, y) \in B_n \times B_n$ ，判断  $x \sim y$  是否成立。根据群表示理论，对于任何群  $G$ ，总存在一个从  $G$  到某一个环的同态，该同态对共轭关系保持不变，因此 CDP 问题对于任何群在计算上总是可解决的。在现有的

- 5 基于辨群共轭问题的签名体制中给出了一个算法可以在多项式时间内以任意高的概率解决 CDP 问题。所谓 CSP 问题就是指：对于一个给定的共轭辨元对  $(x, y) \in B_n \times B_n$  ( $x \sim y$ )，找到一个辨元  $a \in B_n$ ，使得  $y = a^{-1}xa$ 。对于辨群来说，目前不存在一个有效的算法可以在多项式时间内解决 CSP 问题，因此对随机选取的一共轭对  $(x, y) \in B_n \times B_n$ ，他们的 CSP 问题将以很高的概率为一个困难问题。而本文提出的数字签名
- 10 方法的安全性是建立在 MCSP 问题的困难性上的，在现有的基于辨群共轭问题的签名体制中证明了 MCSP 问题与 CSP 问题的困难等价性。所谓 MCSP 问题对他的描述如下：

已知：辨群  $B_n$  的一个共轭对  $(x, x') \in B_n \times B_n$ ，和一个辨元  $y \in B_n$

问题：找到一个  $y' \in B_n$  满足：  $y \sim y'$   $xy \sim x'y'$

15

下面对本发明的数字签名的方法的数学模型做一个简介：

公共参数：辨群  $B_n$ 、左辨群  $LB_m$ 、右辨群  $RB_{n-1-m}$ 、散列函数  $h$ 、其中  $B_n$  的生成元为  $\sigma_1, \sigma_2, \dots, \sigma_{n-1}$ ，左辨群  $LB_m$  为生成元  $\sigma_1, \sigma_2, \dots, \sigma_{m-1}$  生成的  $B_n$  的子群，右辨群  $RB_{n-1-m}$  为生成元  $\sigma_{m+1}, \sigma_{m+2}, \dots, \sigma_{n-1}$  生成的  $B_n$  的子群。

20

密钥生成：公钥：一对 CSP 问题为困难问题的共轭对  $(x, x') \in LB_m \times B_n$  私钥： $a \in B_n$  满足  $x' = a^{-1}xa$ ；

签名：对于一个给定的消息  $m$ ，首先计算  $y = h(m)$ ，然后随机选取一个秘密随机化因子  $b \in RB_{n-1-m}$ ，消息  $m$  的签名  $\text{sign}(m) = a^{-1}byb^{-1}a$ （见图 5）

- 25 验证：对于消息  $m$  签名  $\text{sign}(m)$  为合法签名当且仅当： $\text{sign}(m) \sim y$ ， $x' \text{sign}(m) \sim xy$ （见图 6）

对于一个合法的签名  $\text{sign}(m)$ ，由于  $\text{sign}(m) = a^{-1}byb^{-1}a = (b^{-1}a)^{-1}yb^{-1}a$ ，故  $\text{sign}(m) \sim y$  成立；而  $x' \text{sign}(m) = a^{-1}xa a^{-1}byb^{-1}a = a^{-1}xbyb^{-1}a$ ，由于  $x \in LB_m$ ， $b \in RB_{n-1-m}$ ，因此  $xb = bx$ ，从而有  $x' \text{sign}(m) = a^{-1}xa a^{-1}byb^{-1}a = a^{-1}xbyb^{-1}a = a^{-1}bx yb^{-1}a = (b^{-1}a)^{-1}(xy)(b^{-1}a)$ ，

- 30 从而  $x' \text{sign}(m) \sim xy$ 。因此一个合法的签名总是可以通过验证的。

而对于一个攻击者来说，他要想伪造一个消息  $m$  的签名，所能知道的只包括公钥

$(x, x')$ , 和  $y=h(m)$ , 要想伪造的签名  $\text{sign}(m)$  满足  $\text{sign}(m) \sim y, x' \text{sign}(m) \sim xy$ , 显然等价于解决 MCSP 问题, 因此是不能做到的。

而对于可以截获分析的消息签名对  $(y_i, b^{-1}ay_ia^{-1}b)$ , 由于加入了随机化因子  $b$ , 可以很好的避免  $k$ -CSP 问题。所谓  $k$ -CSP 问题描述如下:

- 5 已知:  $k$  对共轭对  $(x_i, x_i'), \dots, (x_k, x_k') \in B_n \times B_n$  且  $x_i' = a^{-1}x_ia$  ( $i=1 \dots k$ );  
问题: 找到  $b \in B_n$ , 使得  $x_i' = b^{-1}x_ib$  ( $i=1, 2 \dots k$ )。

以上是对本发明签名方法的数学描述, 然而由于辨群是无限群, 为了用计算机实现, 需要设置系统参数。首先设定系统参数  $n, l, d$  (推荐  $l=3, d=4$ )。令  $B_n(l) = \{b \in B_n \mid 0 \leq \inf(b), \sup(b) \leq l\}$ , 则  $|B_n(l)| < (n!)^l$  为有限的。同理  $LB_m(l) = \{b \in LB_m \mid 0 \leq$

- 10  $\inf(b), \sup(b) \leq l\}$ ,  $RB_{n-1-m}(l) = \{b \in RB_{n-1-m} \mid 0 \leq \inf(b), \sup(b) \leq l\}$ 。对于一个辨元采用目前已知在计算机上计算速度最快的 Burau 表示, 即用一个 Laurent 多项式环  $Z[t, t^{-1}]$  上的  $(n-1) \times (n-1)$  阶矩阵来表示, 具体替换原则如下:

做如下的替换:

$$\sigma_1 = \begin{bmatrix} -t & & & \\ 1 & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix} \quad \sigma_2 = \begin{bmatrix} 1 & t & & \\ & -t & & \\ & 1 & 1 & \\ & & & \ddots \\ & & & & 1 \end{bmatrix} \quad \dots \quad \sigma_i = \begin{bmatrix} & & & \\ & 1 & t & \\ & & -t & \\ & & 1 & 1 \\ & & & & \ddots \end{bmatrix}$$

15

$$\sigma_{n-1} = \begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & 1 & t \\ & & & & -t \end{bmatrix}$$

一个属于  $B_n(l)$  辨元转化为一个 Burau 表示的计算复杂度为

$O(\ln)$ , 有了以上表示, 与辨群内的群运算和求逆运算就转化为矩阵的乘法和求逆运算, 他们都有着很有效的数学工具可以解决, 它们的计算复杂度都为  $O(\ln)$ 。

- 20 现在介绍一下 CDP 问题的判定算法 BCDA (见图4)。任何一个非交换群, 都存在一个从群到一个环的函数, 该函数对于共轭对的函数值相等, 把该函数叫做特征。定义一个从  $B_n(l)$  到 Laurent 多项式环  $Z[t, t^{-1}]$  的一个函数:  $g \rightarrow \det(\phi(g) - I)$ , 其中  $g \in B_n(l)$ ,  $\phi(g)$  为  $g$  的 Burau 表示,  $I$  为单位矩阵,  $\det()$  为求行列式的符号, 显然该函数为  $B_n(l)$  的特征。把  $\det(\phi(g) - I)$  叫做辨元  $g$  的亚历山大多项式, 记做  $P_g(t)$ , 显然对于一个  $g \in B_n(l)$ , 它的亚历山大多项式  $P_g(t)$  的秩  $\partial(P_g(t)) \leq l(n-1)n/2$ 。判断两个辨元  $a, b \in B_n(l)$  是否共轭, 做如下的亚历山大测试: 选

定系统参数素数 $p$ 和正整数 $r$ , 在有限域 $\mathbb{Z}/p\mathbb{Z}$ 上任意选取 $r$ 个不相等的值 $t_1, t_2 \dots t_r$ , 若对于所有的 $t_i$  ( $i=1, 2 \dots r$ ) 都有 $P_a(t_i)=P_b(t_i)$ , 则输出1, 否则输出0。由于 $\partial(P_a(t)-P_b(t)) \leq l(n-1)n/2$ , 所以方程 $P_a(t)-P_b(t)=0$ 最多只有 $l(n-1)n/2$ 个根。所以概率 $\Pr[P_a(t) \neq P_b(t) | \text{亚历山大测试输出为1}] \leq \left(\frac{l(n-1)n}{2p}\right)^r$  显然随着 $p$ 和 $r$ 的增加, 这个概率可以任意的小。亚历山大测试的计

5 算复杂度为 $O(rn^3)$ 。

Maxinf-Minsup测试。对于辨元的 $x \in B_n(l)$ , 定义 $\text{Maxinf}(x) = \text{Max}\{\inf(y) | y \sim x, y \in B_n(l)\}$ ,  $\text{Minsup}(x) = \text{Min}\{\sup(y) | y \sim x, y \in B_n(l)\}$ 。所谓Maxinf-Minsup测试, 即对辨元 $a, b \in B_n(l)$ , 判断 $\text{Maxinf}(a) = \text{Maxinf}(b)$ ,  $\text{Minsup}(a) = \text{Minsup}(b)$ 是否都成立, 若都成立, 则输出1, 否则输出0。下面给出计算 $\text{Maxinf}(x)$ 和 $\text{Minsup}(x)$ 的算法描述。首先定义

- 10 两个操作, 若 $x = \Delta^u \pi_1 \pi_2 \dots \pi_l$ ,  $\text{cycling}(x) = (\tau^u(\pi_l))^{-1} x \tau^u(\pi_l)$ ,  $\text{decycling}(x) = \pi_l^{-1} x \pi_l$ 。对 $x$ 循环做 $\text{cycling}$  (分别  $\text{decycling}$ ) 操作, 直到 $\inf$ 值增加 (分别  $\sup$ 值减少), 然后以当前得到的元素为新元素, 重复该循环操作, 且循环次数计数重新设置为1; 若循环次数计数直到 $m=n(n-1)/2$ 都 $\inf$ 值不再增加 (分别  $\sup$ 值不再减少), 则当前的元素的 $\inf$ 值即为 $\text{Maxinf}(x)$  (分别  $\text{Minsup}(x)$ )。该算法的理论分析请参看引文: *J. S. Birman, K. H. Ko and S. J. Lee, The infimum, supremum and geodesic length of a braid conjugacy class, to appear in Advances in Mathematics.*。该算法的算法复杂度为 $O(l^2 n \log n)$ 。

如果辨元 $a, b$ 通过了亚历山大测试和Maxinf-Minsup测试, 则可以判定 $a \sim b$ 成立, 只有一种例外情况, 即 $a \sim b^{-1}$ 。然而这对于随机选择的 $a, b$ 来说, 是几乎不可能出现的, 而对于攻击者同样不能利用这种例外情况, 分析见引文: *K. H. Ko, S. J. Lee, J. H. Cheon, J. W. Han, J. S. Kang and C. S. Park, New Public-Key Cryptosystem Using Braid Groups, Proc. of Crypto 2000, LNCS 1880, Springer-Verlag (2000) 166 # 183.*

- 20 *W. Han, J. S. Kang and C. S. Park, New Public-Key Cryptosystem Using Braid Groups, Proc. of Crypto 2000, LNCS 1880, Springer-Verlag (2000) 166 # 183.*

对于一个从比特串 $\{0, 1\}^*$ 到辨群 $B_n(l)$ 的散列函数 $h$ , 可构造如下 (见图3), 首先我们使用一个普通散列函数将 $\{0, 1\}^*$ 压缩得到固定长度的比特串 $\{0, 1\}^N$ , 其中 $N = \lceil l \log_2 n \rceil$ 。然后将 $\{0, 1\}^N$ 分为 $l$ 段 $r_1 \| r_2 \| \dots \| r_l$ , 每一段的长度都为 $\lceil \log_2 n \rceil$ 。由于 $B_n(l)$ 的置换元有 $n!$ 个,

- 25 故可在置换元和整数集 $[0, n!-1]$ 间建立一个一一映射。因此再依次将 $r_k$ 转化为 $[0, n!-1]$ 间某一个整数, 再将这个整数和一个置换元 $P_k$ , 最后得到 $h(m) = \prod_{k=1}^l P_k$ 。

为了产生一个伪随机辨元, 首先产生一个伪随机序列 $\{0, 1\}^N$ , 然后将 $\{0, 1\}^N$ 分为 $l$ 段

$r_1 \| r_2 \| \cdots \| r_l$ , 每一段的长度都为  $\lceil \log_2 n \rceil$ 。由于  $B_n(l)$  的置换元有  $n!$  个, 故可在置换元和整数集  $[0, n!-1]$  间建立一个一一映射。因此再依次将  $r_k$  转化为  $[0, n!-1]$  间某一个整数, 再将这个整数和一个置换元  $P_k$ , 最后得到所需要的伪随机置换元  $= \prod_{k=1}^l P_k$ 。

5 为了安全的产生密钥, 先定义一些概念, 对于一个置换元  $x \in B_n(l)$ , 定义他的 super summit 集  $SSS(x) = \{y \in B_n(l) | y \sim x, \inf(y) = \text{Maxinf}(x), \sup(y) = \text{Minsup}(x)\}$ 。整个签名算法的安全强

度为  $|SSS(x)|$ , 约为  $\left(\frac{n}{4}\right)^{n(n-1)/2}$ 。若  $y \sim x$ , 定义  $x, y$  间的距  $d(x, y) = \min\{l(b) | y = b^{-1}ab\}$ , 然后

再定义  $s(x, d) = \{y \in SSS(x) | d(x, y) \leq d\}$ 。选取  $x' \in s(x, d)$ , 则共轭对  $(x', x)$  的 CSP 问题为困难问题, 可以作为密钥, 下面介绍算法  $RSSBG(x, d) = (x', a)$  用以随机产生  $x' \in s(x, d)$  且  $x' = a^{-1}xa$ , 从而得到安全公钥  $(x', x)$  和私钥  $a$  (见图2):

- 10 1 随机产生一个置换元  $b \in B_n(5l)$ , 计算  $b^{-1}xb$ 。
- 2 随机选择对  $b^{-1}xb$  一系列的 cycling 或者 decycling 操作得到  $x'$ , 直到  $x'$  属于  $SSS(x)$  为止 (不会超过  $n^2$  次操作),  $x' = a^{-1}xa$ 。
- 3 判断  $l(a) \leq d$  成立, 若成立, 则  $x'$  和  $a$  即为所求, 若不成立, 则转1。

15 有了以上的数学基础和算法描述, 再结合附图对技术方案的实施作进一步的详细描述, 本发明用软件实现如下 (为了提高速度, 算法 BCDA 也可用硬件实现):

选定系统参数公开:

置换群参数  $n, l, d, p$  (推荐  $n$  为  $20 \sim 30$  间,  $l=3, d=4, p$  为  $2^{31} \sim 2^{32}$  间), 以及左置换群大小  $m$  (推荐  $n-m$  为 4 左右)。

20 选定用于散列消息的散列函数  $h$ :

参考图1所示的流程,

签名方  $S$ :

密钥产生:

- 1 使用算法 PBG 随机产生一个置换元  $x \in LB_m$ ;
- 25 2 使用算法  $RSSBG(x, d)$  得到公钥  $(x', x)$  和私钥  $a$ 。

签名:

- 1 对需要签名的消息  $m$  应用散列函数  $h$  得到  $y = h(m)$ ;
- 2 随机产生一个置换元  $b, x \in a^{-1}byb^{-1}a$ , 然后计算  $byb^{-1}$ ;
- 3 使用私钥, 计算签名  $\text{sign}(m) = a^{-1}byb^{-1}a$ 。

验证方V:

- a) 对需要验证签名的消息 $m$ 应用散列函数 $h$ 得到 $y=h(m)$ ;
- b) 使用算法BCDA判定 $\text{sign}(m) \sim y$ 是否成立, 若不成立, 则验证失败, 结束。若成立, 转3;
- 5 c) 计算 $x'\text{sign}(m)$ 和 $xy$ ; 使用算法BCDA判定 $x'\text{sign}(m) \sim xy$ 是否成立, 若成立, 则验证通过, 结束, 否则验证失败, 结束。

# 说明书附图

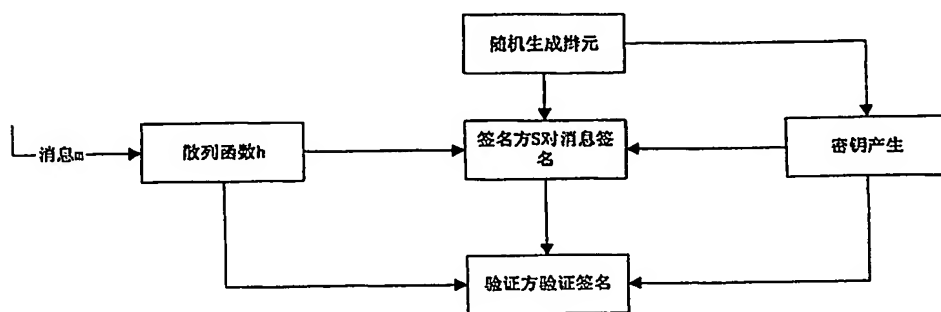


图 1

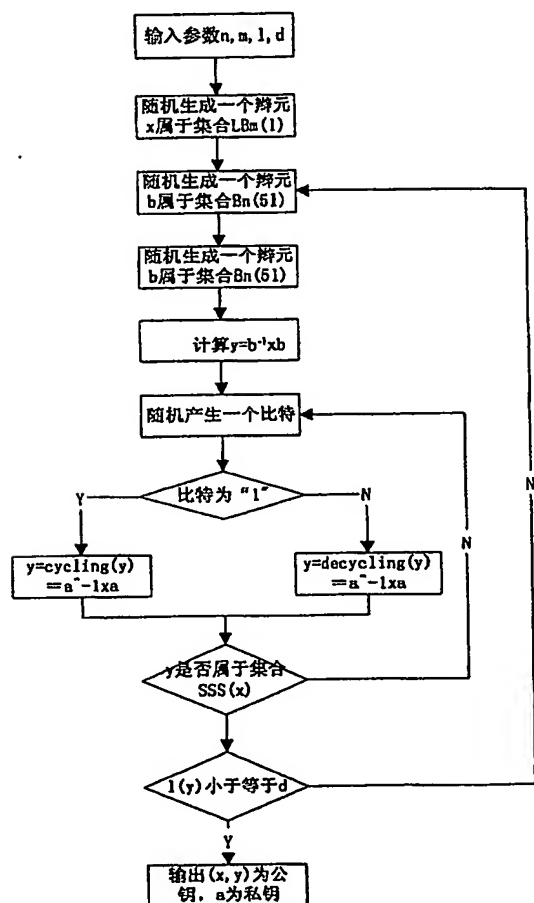


图 2

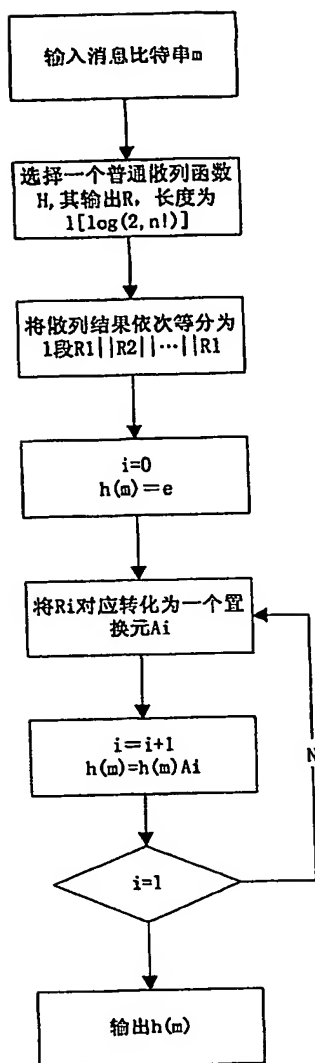


图 3

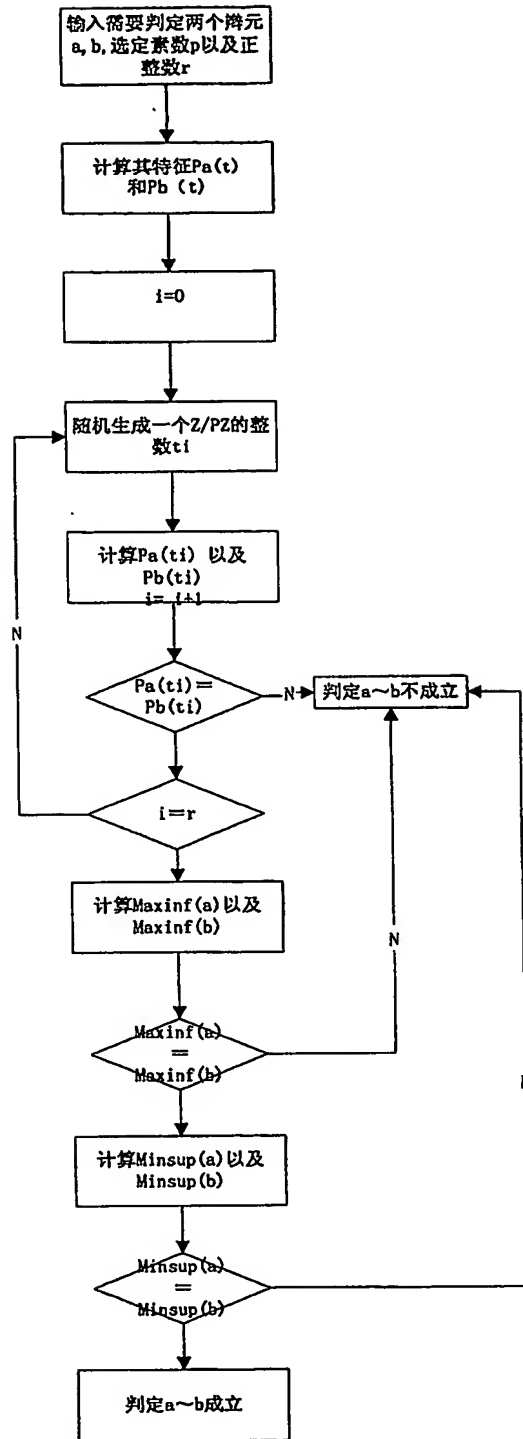


图 4



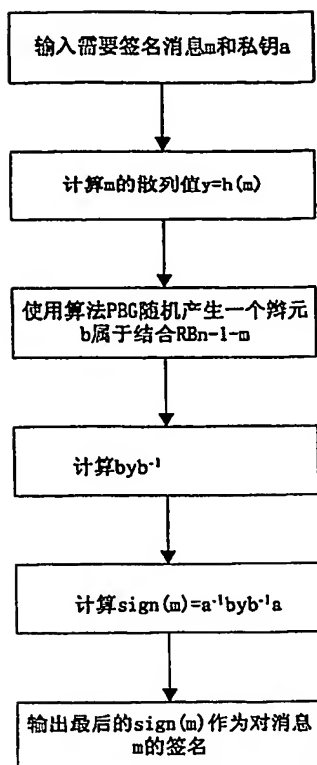


图 5

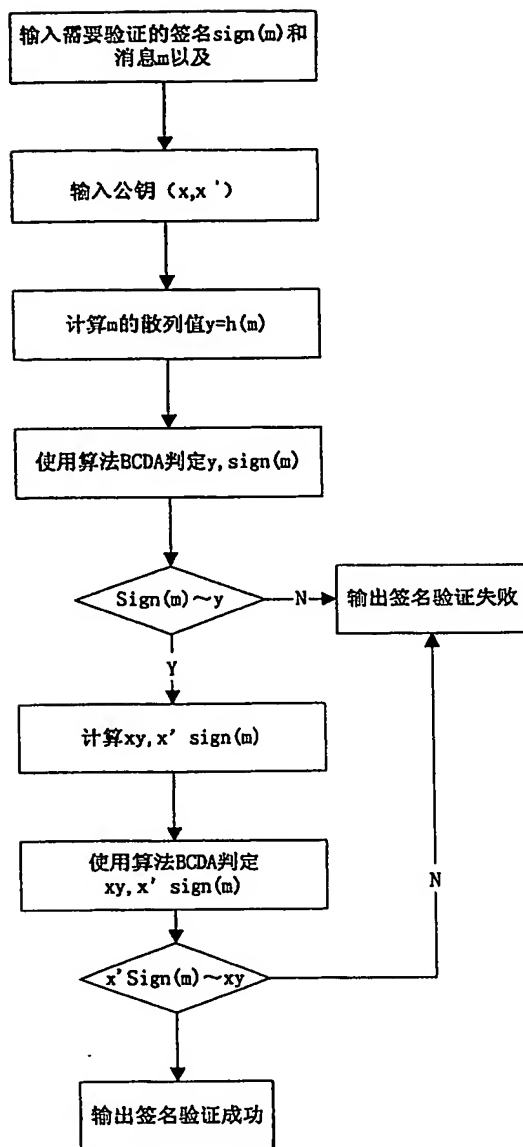


图 6

# Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/CN04/001289

International filing date: 12 November 2004 (12.11.2004)

Document type: Certified copy of priority document

Document details: Country/Office: CN  
Number: 200310113604.9  
Filing date: 13 November 2003 (13.11.2003)

Date of receipt at the International Bureau: 26 January 2005 (26.01.2005)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b)



World Intellectual Property Organization (WIPO) - Geneva, Switzerland  
Organisation Mondiale de la Propriété Intellectuelle (OMPI) - Genève, Suisse

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☒ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**